Journal of Computational and Applied Mathematics 235 (2011) 4639–4651

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

CMARS and GAM & CQP—Modern optimization methods applied to international credit default prediction

Özge Sezgin Alp [a,b], Erkan Büyükbebeci [c], Ayşegül İşcanoğlu Çekiç [a,d], Fatma Yerlikaya Özkurt [a], Pakize Taylan [e], Gerhard-Wilhelm Weber [a,*]

[a] Institute of Applied Mathematics, Middle East Technical University, 06531 Ankara, Turkey
[b] Faculty of Commercial Sciences, Başkent University, 06530 Ankara, Turkey
[c] Turkish Statistical Institute, 06100 Ankara, Turkey
[d] Department of Mathematics, Selçuk University, 42697 Konya, Turkey
[e] Department of Mathematics, Dicle University, 21280 Diyarbakır, Turkey

ARTICLE INFO

Article history:
Received 26 December 2009
Received in revised form 18 April 2010

Keywords:
Financial mathematics
Sovereign defaults
Emerging markets
CART
GAM
Logistic regression
Regularization
MARS
CMARS
Continuous optimization
Conic quadratic programming

ABSTRACT

In this paper, we apply newly developed methods called GAM & CQP and CMARS for country defaults. These are techniques refined by us using Conic Quadratic Programming. Moreover, we compare these new methods with common and regularly used classification tools, applied on 33 emerging markets' data in the period of 1980–2005. We conclude that GAM & CQP and CMARS provide an efficient alternative in predictions. The aim of this study is to develop a model for predicting the countries' default possibilities with the help of modern techniques of continuous optimization, especially conic quadratic programming. We want to show that the continuous optimization techniques used in data mining are also very successful in financial theory and application. By this paper we contribute to further benefits from model-based methods of applied mathematics in the financial sector. Herewith, we aim to help build up our nations.

1. Introduction

With globalization, international loans and country default probabilities have become one of the main topics of finance. In the last decade of 20th century, many important economic crises occurred in emerging markets. Therefore, the probabilities of countries not meeting their external debt obligations has attracted the interest of scientists. In the literature there are many studies that use quantitative methods to predict countries' defaults or, in other words, countries rescheduling.

In 1971, Frank and Cline [1] used linear and quadratic discriminant analysis to determine the rescheduling abilities of countries. In 1977, Feder and Just [2] tried to determine debt servicing capacities of countries and rescheduling probabilities with logistic regression. They tested the predictability and found a relatively low rate of errors. In 1979, Fisk and Rimlinger [3] studied long run credit worthiness and rescheduling of developing countries with a probit model by using economic indicators.

* Corresponding author. Tel.: +90 312 210 5652; fax: +90 312 210 2985.
E-mail addresses: osezgin@baskent.edu.tr (Ö.S. Alp), erkan.buyukbebeci@tuik.gov.tr (E. Büyükbebeci), iaysegul@metu.edu.tr (A.İ. Çekiç), fatmayerlikaya@gmail.com (F.Y. Özkurt), ptaylan@dicle.edu.tr (P. Taylan), gweber@metu.edu.tr (G.-W. Weber).

0377-0427/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2010.04.039

In 1984, Taffler and Abassi [4] used linear discriminant analysis and bankers' judgements for predicting developing countries' debt rescheduling with monetary policy and debt servicing indicators. They built an early warning system with discriminant analysis and got better results than judgmental methods.

In 1992, Balkan [5] used a probit model to predict default and concluded that a model with political instability and economic variables has better results than a model with economic variables only. In 1997, de Haan et al. [6] also built a probit model for debt rescheduling with economic and political indicators. In 1997, Rivoli and Brewer [7] built a logistic regression model for countries and they discussed the measurement of political indicators. In 1999, Cooper [8] used the Artificial Neural Network (ANN) model to identify countries likely to seek a rescheduling of their international debt, compared the results with multivariate statistical procedures and concluded that ANN is very useful for default problems. In 1999, Easton and Rockerbie [9] built a tobit model to obtain the default probabilities of sovereign debts and compared the results with logistic regression. They found that tobit scaled probabilities outperform the other model.

In 2006, Gestel et al. [10] built a generic process model by using ordinal logistic regression with kernel based learning capability to build an internal rating system for sovereign debt. In 2009, Maltritz et al. [11] studied the country default risk with a multiple indicator multiple causes approach. So they were able to consider several indicators of default as dependent variables at once.

In 2006, Fioramanti [12] used artificial neural networks to build an early warning system in predicting a sovereign debt crisis. He showed that when the number of hidden units, training epochs and efficient training algorithm are chosen correctly, ANN results are at least as good as those of parametric traditional methods.

The aim of this study is to develop a model for predicting the countries' default possibilities with the help of new techniques of modern continuous optimization, especially, conic quadratic programming. CMARS method was developed as an alternative to MARS, which is modified by constructing a penalized residual sum of squares (PRSS) as a Tikhonov regularization problem and solving this problem by using Conic Quadratic Programming (CQP). On the other hand, for Generalized Additive Models (GAM), the optimization problem is built up with smoothed functions as base-splines and this optimization problem is resolved by CQP. In addition, the new methods (CMARS and GAM & CQP) are compared with mostly used statistical classification techniques. We intend to give a contribution to the theory by using newly developed techniques with optimization. We want to show that the continuous optimization techniques used in data mining are also very successful in financial theory and application.

The rest of the paper is organized as follows. In Section 2, we give the classification methods and validation techniques used in the analysis part. Then, in Section 3 we explain the data and give application results. Finally, in Section 4, we give a conclusion and an outlook on future studies.

2. Methodology

In this section methods that will be used for default prediction are given. Default prediction is a classification problem that tries to determine whether the countries would default on international loans or not by using some indicators. In other words, it is for building a model that helps to predict countries' future performances by using the past information about their default behaviours. In the models, there are independent and dependent variables. Here, the dependent variable shows if the country is defaulted or not; the independent variables include indicators about countries.

2.1. Classification techniques

2.1.1. Logistic regression

In sovereign rating studies, binary logistic regression is applied since the dependent variable is binary. This regression is a part of statistical models, especially of generalized linear models. In logistic regression, the maximum likelihood estimation is used after applying Logit transformation to the dependent variable. In binary logistic regression, the model is defined as:

$$E[Y|\boldsymbol{x}] = P\{Y = 1|\boldsymbol{x}\} = F\left(\boldsymbol{x}^T\boldsymbol{\beta}\right) = \frac{1}{1 - \exp\left(-\boldsymbol{x}^T\boldsymbol{\beta}\right)} = \pi. \tag{1}$$

Here, F is the cumulative distribution function (inverse link function), $\boldsymbol{\beta}$ is the unknown parameter vector of the model, and π is the probability that the dependent variable takes the value 1.

Since we are employing binary logistic regression, the dependent variable is assumed to distribute as Bernoulli. To estimate the unknown parameters it is necessary to write the likelihood function by using the Bernoulli assumption:

$$L(\boldsymbol{x}_i) := \prod_{i=1}^{N} \pi(\boldsymbol{x}_i)^{y_i}(1 - \pi(\boldsymbol{x}_i))^{1-y_i} \quad (i = 1, \ldots, N), \tag{2}$$

where $\pi(\boldsymbol{x}_i)$ is the probability of each observation with independent variable vector $\boldsymbol{x}_i$, that takes the value 1 as dependent variable.

Since, mathematically, it is easier to maximize the natural logarithm of the likelihood function, the log-likelihood function is as follows

$$l(\boldsymbol{x}_i) = \sum_{i=1}^{n} \{y_i \ln(\, p_i(\boldsymbol{x}_i)) + (1 - y_i) \ln(1 - p_i(\boldsymbol{x}_i))\}. \tag{3}$$

The estimate of the unknown parameters $\hat{\boldsymbol{\beta}}$ is obtained by solving the following equation:

$$\frac{\partial \ln L(\hat{\boldsymbol{\beta}})}{\partial \boldsymbol{\beta}} = \mathbf{0}. \tag{4}$$

Generally, it is solved by iterative optimization methods. The model can be tested after unknown parameters are estimated. The significant logistic model may be applied to predict future values of observations.

2.1.2. Classification and regression trees

Classification and Regression Trees (CART) is a non-parametric method of classification. It was first introduced in [13]. In this method, the observations are split into different classes by binary decision rules.

There are two types of trees: classification tree and regression tree. When classifying data into classes, a classification tree is used, and if the dependent variable is continuous, a regression tree is used. In our sovereign credit rating problem, the dependent variable is either default or non-default, so classification tree is suitable for our case.

In classification tree, there are three basic steps:

- Constructing the maximum sized tree,
- Determining the right sized tree, and
- Classifying the new observations by using the right sized tree.

The CART tree is the combination of nodes. The nodes include binary decisions and the sample is divided into two subsets according to these binary decision rules. Then, the first step in constructing the tree is to determine the proper variable right question rule for each node. This is called the *splitting* of the sample. The main problem in constructing the tree is to determine how to split the sample. The steps of splitting are given below:

i. All the observations of an independent variable are placed in the root node.
ii. For each observation x_{ij} of independent variable x_j, the standard question $x_j \leq x_{ij}$ is asked and observations that satisfy the condition are placed to the left child node and the others are placed to the right child node.
iii. For each splitting decision, the goodness of split criteria are estimated and according to the criteria the best rule is selected. The best rule is the one that splits the data homogeneously.
iv. The process is repeated for all independent variables.

Another important aspect in constructing the maximum sized tree is to decide where to stop splitting. Therefore, it is necessary to select a node as terminating or non-terminating by checking the termination rule. The last step in constructing the maximum sized tree is to assign a class to each terminal node by means of class assignment rules. The process continues until each observation is assigned to a class.

Goodness of split criteria: As mentioned before, the best splitting decision attains the homogeneity of each child node. Homogeneity of the child node is called the *impurity*. It is a special function defined as follows:

$$i(t) = \phi\,(p(1|t), \ldots, p(c|t))\,, \tag{5}$$

where

$$p(c|t) = \frac{p(c, t)}{p(t)}, \tag{6}$$

$$p(t) = \sum_{c=1}^{C} p(c, t), \tag{7}$$

$$p(c, t) = \pi(c)\frac{N_c(t)}{N_c}, \tag{8}$$

$$\pi(c) \coloneqq \frac{N_c}{N}. \tag{9}$$

Here, N_c is the number of observations belong to the class c, N is the number of observations, $\pi(c)$ is the prior probability of class c, $N_c(t)$ is the number of observations in node t belonging to class c, $p(c, t)$ is the probability of an observation both in class c and in node t, and $p(c|t)$ is the probability of class c in node t. The impurity function has the following special properties:

i. When $p(1|t) = p(2|t) = \cdots = p(c|t) = \frac{1}{c}$ the impurity function $\phi\,(\frac{1}{c}, \ldots, \frac{1}{c})$, takes its maximum value.
ii. In a node t, when there is only one majority of class, then the impurity function $\phi(p(1|t), \ldots, p(c|t))$ takes its minimum value.
iii. ϕ is a symmetric function of probability of class c in node t.

The best splitting rule is the one with maximum change of impurity. The best known criteria are the Gini Index and the Twoing rule.

The *Gini impurity* function is linear and quadratic. Its change of impurity function is given by

$$i(t) = \sum_{i \neq j} p(j|t)p(i|t), \tag{10}$$

$$\Delta i(t) = i(t) - p_L i(t_L) - p_R i(t_R), \tag{11}$$

where t_L and t_R are left and right child nodes, respectively. In the *Twoing rule*, for each node the impurity is estimated as a binary problem. Its change of impurity function is

$$\Delta i(t) = \frac{p_L p_R}{4} \left[\sum_{c=1}^{C} |p(c/t_L) - p(c/t_R)| \right]^2. \tag{12}$$

The terminating rule: In each node, when determining whether to stop or continue, it is necessary to check if it is impossible to increase the homogeneity of the tree. The homogeneity of the tree is determined by the change of impurity of the tree and it is defined as

$$\Delta I(T) = \Delta i(t)p(t). \tag{13}$$

Here, $p(t)$ is the probability of any observation belonging to node t.

The class assignment rule: There are two ways of assigning a class to a terminated node. The first is to assign a class that has the highest probability. The second is to assign a class which minimizes the expected misclassification cost which is defined as

$$r(t) := \sum_{c=1}^{C} cost(d|c)p(c|t), \tag{14}$$

where $cost(d|c)$ is the cost of assigning a class d to a class c observation.

After class assignment, the maximum sized tree is formed but it is not the optimal tree because it is too complex and not accurate. Therefore, it is necessary to determine the right sized tree. The right sized tree is determined by a pruning process. This is the process of forming all subtrees of a maximum sized tree. In that process, sub-trees are formed by deleting some nodes from the maximum sized tree. Then the cost of complexity measures of the sub-trees is estimated. The cost of complexity measure is defined as:

$$R_\alpha(T) = R(T) + \alpha \tilde{T}; \tag{15}$$

here, $T \leq T_{\max}$, $R(T)$ is the misclassification cost of the tree, $\alpha \geq 0$ is the complexity (tuning) parameter. It is also known as the penalty of additional terminal nodes, and $\tilde{T}$ is the number of terminal nodes of the tree. The sub-tree with minimum $R_\alpha(T)$ is taken as the right sized tree.

2.1.3. Generalized additive models

Generalized Additive Models (GAM) are in common sense applications of several nonparametric and parametric techniques developed in [14].

Let N be the number of observations and p be the number of explanatory variables. The generalized additive models are defined as follows

$$G(\mu(\boldsymbol{x})) = y_i = \beta_0 + \sum_{j=1}^{p} f_j(x_{ij}) \quad (i = 1, 2, \ldots, N), \tag{16}$$

where the mappings f_j are unspecified smoothed functions estimated from the data, μ stands for the expected value and G is the link function. The classical attention to these functions is based on the assumption that $E\left(f_j(x_j)\right) = 0$ $(j = 1, 2, \ldots, N)$ [14]. In the setting of GAM, it is allowed that the dependent variable not to be Gaussian and not necessarily continuous. This is an important feature of GAM because it allows us to construct more flexible models in the estimations. Advantages of GAM can be listed as follows [15]:

- Not only parametric models but also nonparametric and semiparametric additive models can be estimated by GAM.
- GAM can work with large dimensions of determinants.
- By specifying the link function (G) it can work on discontinuous dependent variables with multiple entity.

The functions f_i employed in literature are usually splines and they are selected by preserving density and variability properties of the data set. In this paper, we use the approach expressed in [16]. In classical sense, the regularized least-

squares problem expressed as a penalized problem firstly [17] can be stated as

$$\begin{aligned} \min_{f} \quad & \sum_{i=1}^{N}\left\{y_i-\beta_0-\sum_{j=1}^{p}f_i(x_{ij})\right\}^2 \\ \text{subject to} \quad & \int\left[f_j''(t_j)\right]^2\,\mathrm{d}t_j \le M_j \quad (j=1,2,\ldots,p), \end{aligned} \tag{17}$$

where x_{ij} are the components of the input data, and $f := (f_1, f_2, \ldots, f_N)^T$ consists of our model functions from a suitable function space. The integration, symbolized by the dummy variable "$\int$", takes place over suitable intervals in the corresponding input dimensions. In [16], the problem (17) is converted to the following *Conic Quadratic Programming (CQP)* problem:

$$\begin{aligned} \min_{t,\beta_0,f} \quad & t \\ \text{subject to} \quad & t \ge 0, \\ & \sum_{i=1}^{N}\left\{y_i-\beta_0-\sum_{j=1}^{p}f_i(x_{ij})\right\}^2 \le t^2, \\ & \int\left[f_j''(t_j)\right]^2\,\mathrm{d}t_j \le M_j \quad (j=1,2,\ldots,p). \end{aligned} \tag{18}$$

In analysis, the unspecified smoothed functions include a convex combination of d_j base-splines, which is

$$f_j(x)=\sum_{k=1}^{d_j}\theta_k^j h_k^j(x) \quad (k=1,2,\ldots,d_j), \tag{19}$$

where $h_k^j:\mathbb{R}\to\mathbb{R}$ presents the kth base-spline of variable x_j, and θ_k^j is the corresponding coefficient. In estimation, we used the discretized version of the problem (18). Now, the approximated version of the problem is of the following form:

$$\begin{aligned} \min_{t,\beta_0,\boldsymbol{\theta}} \quad & t \\ \text{subject to} \quad & t \ge 0, \\ & \sum_{i=1}^{N}\left\{y_i-\beta_0-\sum_{j=1}^{p}\sum_{k=1}^{d}\theta_k^j h_k^j(x_{ij})\right\}^2 \le t^2, \\ & \sum_{i=1}^{N-1}\left\{\sum_{k=1}^{d_j}\theta_k^j\omega_i h_k^{j''}(x_{ij})\right\} \le M_j \quad (j=1,2,\ldots,p). \end{aligned} \tag{20}$$

Here, the vector $\boldsymbol{\theta}$ comprises all the coefficients θ_k^j, and $\omega_i=\sqrt{x_{i+1j}-x_{ij}}$.

2.1.4. Multivariate adaptive regression splines

Multivariate Adaptive Regression Splines (MARS), an implementation of techniques developed in [18], is an adaptive regression procedure for solving high dimensional problems when there are many explanatory variables. MARS is very useful in high dimensions and shows great promise for fitting nonlinear multivariate functions with applicability in many areas of finance, science and technology [19]. MARS builds models by fitting piecewise linear basis functions (BFs) in where both the additive and the interactive effects of the explanatory variables are allowed to determine the dependent variable [20].

MARS uses expansions of the piecewise linear basis function, $(x-t)_+$ and $(t-x)_+$ with a knot value at t. The following two functions, where $x\in\mathbb{R}$ [21], are truncated:

$$(x-t)_+ := \begin{cases} x-t, & \text{if } x>t, \\ 0, & \text{otherwise,} \end{cases} \tag{21}$$

$$(t-x)_+ := \begin{cases} t-x, & \text{if } x<t, \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

The symbol '+' means that only the positive parts are used, and are otherwise zero. These two functions are named as a reflected pair. The main idea is to construct reflected pairs for each input x_j with knots at each observed value $x_{i,j}$ $(i=1,2,\ldots,N;\ j=1,2,\ldots,p)$ of that input. Hence, the collection of the BFs is [22]:

$$C := \left\{(x_j-t)_+, (t-x_j)_+ \mid t\in\left\{x_{1,j}, x_{2,j}, \ldots, x_{N,j}\right\}, j\in\{1,2,\ldots,p\}\right\}. \tag{23}$$

If each of the input values is separate, there are $2Np$ BFs totally, where every BF is only related to a single x_j [22].

BFs are the tensor products of univariate spline functions. We apply BFs in order to generalize spline fitting in higher dimensions. Then, multivariate splines BFs take the following form:

$$B_m(\boldsymbol{x}) := \prod_{j=1}^{K_m} [s_{km} \cdot (x_{v(km)} - t_{km})]_+, \tag{24}$$

where K_m is the total number of truncated linear functions in the mth BF, $x_{v(km)}$ is the input variable corresponding to the kth truncated linear function in the mth BF, t_{km} is the corresponding knot value and $s_{km} \in \{\pm 1\}$ [20].

Let Y be a continuous or a binary dependent random variable, $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^T$ be a vector of independent variables showing the following input-response relationship:

$$Y = f(\boldsymbol{x}) + \epsilon, \tag{25}$$

where $f(\boldsymbol{x})$ is the model function, and ϵ is the error term in the observation.

The MARS procedure for estimating the model function $f(\boldsymbol{x})$ consists of two algorithms [18]. The first algorithm is analogous to forward stepwise linear regression and MARS allows the use of basis functions and their products from the set C constructing the maximal model to overfit data initially. The form of the model is as follows:

$$f(\boldsymbol{x}) = \beta_0 + \sum_{m=1}^{M} \beta_m B_m(\boldsymbol{x}), \tag{26}$$

and each $B_m(\boldsymbol{x})$ is a function from the set C, or the product of two or more functions from C, M is the number of BFs in the current model and β_0 is the intercept [22,23].

By standard linear regression, given some choices for the B_m, the coefficients β_m are estimated by minimizing the residual sum of squares. To generate the model, the crucial point is the construction of the B_m. The forward stepwise algorithm searches for the basis function and at each step, the split that minimized some *lack of fit* criterion from all the possible splits on each basis function is chosen. The process stops when a user-specified value $M_{\max}$ is reached [18,20].

The maximal model normally overfits the data. In order to avoid this, we start a backward elimination process. In each step we delete the term which causes the smallest increase in the residual squared error. We continue until we attain the optimally estimated model $\hat{f}_\alpha$ with the optimal value α.

In order to estimate the optimal value of α reducing computational costs, *generalized cross-validation (GCV)* is used. This formulation, also known as the lack of fit criterion, is defined as [24]

$$GCV_{Friedman} := \frac{\sum_{i=1}^{N} (y_i - \hat{f}_\alpha(\boldsymbol{x}_i))^2}{\left(1 - \frac{\tilde{B}(\alpha)}{N}\right)^2}, \tag{27}$$

where N is the number of data samples, and $\tilde{B}(\alpha)$ is the number of valid parameters being fit. The numerator of $GCV_{Friedman}$ is the known common residual sum of squares, which is penalized by the denominator. This denominator accounts for the increasing variance in the case of increasing the model complexity [20].

Here, $\tilde{B}(\alpha) = r + cK$ with K representing the number of knots which are selected in the forward process and r is the number of linearly independent functions in the model. The quantity c shows a cost for each BF optimization and, generally, $c = 3$ [20,25]. Using the lack of fit criteria, we choose the best model according to the backward sequence that minimizes $GCV_{Friedman}$ [20,23,25].

The MARS models in this study are fitted using MARS (Version 2, Salford Systems, San Diego, Calif., USA). The MARS package developed by Salford Systems is available under [21].

2.1.5. CMARS

As mentioned in the previous section, the MARS procedure for estimating the model function consists of two algorithms; these are the *forward* and the *backward stepwise algorithms*. On the other hand, instead of using the backward stepwise algorithm, we construct a penalized residual sum of squares for MARS as a *Tikhonov regularization* problem. For solving this problem, we use one of the modern continuous optimization techniques, which is *conic quadratic programming (CQP)* [20]. By this we arrive at a new alternative method, named CMARS ("C" standing for convex, but also reminding of continuous and conic). For explanations on CMARS, the following notation for the piecewise linear BFs is preferred:

$$c^+(x, \tau) = (+(x - \tau))_+, \qquad c^-(x, \tau) = (-(x - \tau))_+, \tag{28}$$

where $(q)_+ := \max\{0, q\}$ and τ is a univariate knot [20].

Recall the notation given in [25] and for technical reasons reformulate our set of BFs as follows now:

$$\wp := \left\{(x_j - \tau)_+, (\tau - x_j)_+ | \tau \in \left\{x_{1,j}, x_{2,j}, \ldots, x_{N,j}\right\},\ j \in \{1, 2, \ldots, p\}\right\}. \tag{29}$$

Thus, we can represent $f(\boldsymbol{x})$ by a linear combination which is successively built up by the set $\wp$ and with the intercept θ_0, so that our model takes the form

$$Y = \theta_0 + \sum_{m=1}^{M} \theta_m \psi_m(\boldsymbol{x}) + \epsilon, \tag{30}$$

where M is the number of BFs suitable for the data [20]. Here, ψ_m $(m = 1, 2, \ldots, M)$ represents a BF from $\wp$ or products of two or more such functions, ψ_m is taken from a set of M linearly independent basis elements, and θ_m is the unknown coefficient for the mth BF $(m = 1, 2, \ldots, M)$ for the constant 1, m equals to zero [20].

Provided the observations represented by the data $(\boldsymbol{x}_i, y_i)$ $(i = 1, 2, \ldots, N)$, the form of the mth basis function is as follows [20]:

$$\psi_m(\boldsymbol{x}) := \prod_{j=1}^{K_m} [s_{\kappa_j^m} \cdot (x_{\kappa_j^m} - \tau_{\kappa_j^m})]_+, \tag{31}$$

where K_m is the number of truncated linear functions multiplied in the mth basis function, $x_{\kappa_j^m}$ is the input variable corresponding to the jth truncated linear function in the mth basis function, $\tau_{\kappa_j^m}$ is the knot corresponding to the variable $x_{\kappa_j^m}$, and $s_{\kappa_j^m}$ is the selected sign $+1$ or -1.

As mentioned in the previous section, for estimating the optimal value of α, which shows the complexity of estimation, GCV is used in MARS. This formulation is given in (27). Now instead of running the second part of the MARS algorithm, we add penalty terms to the least squares estimation for a control of the lack of fit. This is an alternative point of view of the complexity and the stability of the estimation [20], and we shall explain it now.

2.1.6. Tikhonov regularization and the construction of the conic quadratic programming

For the MARS model, *penalized residual sum of squares (PRSS)* has the following form:

$$PRSS := \sum_{i=1}^{N} (y_i - f(\boldsymbol{x}_i))^2 + \sum_{m=1}^{M_{\max}} \lambda_m \sum_{\substack{|\boldsymbol{\alpha}|=1 \\ \boldsymbol{\alpha}=(\alpha_1,\alpha_2)^T}}^{2} \sum_{\substack{r<s \\ r,s\in V_m}} \int \theta_m^2 \left[D_{r,s}^{\boldsymbol{\alpha}} \psi_m(\boldsymbol{t}^m)\right]^2 \mathrm{d}\boldsymbol{t}^m, \tag{32}$$

where $V_m := \{\kappa_j^m | j = 1, 2, \ldots, K_m\}$ is the variable set associated with the mth basis function, ψ_m, $\boldsymbol{t}^m = (t_{m_1}, t_{m_2}, \ldots, t_{m_{K_m}})^T$ represents the vector of variables which contribute to the mth basis function, ψ_m. Moreover, $D_{r,s}^{\boldsymbol{\alpha}} \psi_m(\boldsymbol{t}^m) := \frac{\partial^{\boldsymbol{\alpha}} \psi_m}{\partial^{\alpha_1} t_r^m \partial^{\alpha_2} t_s^m}(\boldsymbol{t}^m)$ for $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)^T$, $|\boldsymbol{\alpha}| := \alpha_1 + \alpha_2$, where $\alpha_1, \alpha_2 \in \{0, 1\}$. The integrations, symbolized by the sign "$\int$", take place over suitably chosen measurable domains, say, parallel pipes, in corresponding Euclidean spaces. Our optimization problem is based on the trade-off between both accuracy, i.e., a small sum of error squares, and not too high a complexity. This trade-off is established through the penalty parameters λ_m [20].

From [19,20] it follows that PRSS can be approximated and defined in this way:

$$PRSS \approx \left\| \boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta} \right\|_2^2 + \sum_{m=1}^{M_{\max}} \lambda_m \sum_{i=1}^{(N+1)K_m} L_{im}^2 \theta_m^2, \tag{33}$$

where $\boldsymbol{y} := (y_1, y_2, \ldots, y_N)^T$ is the data vector of responses,
$\boldsymbol{\psi}(\tilde{\boldsymbol{d}}) = \left(\boldsymbol{\psi}(\tilde{\boldsymbol{d}}_1), \ldots, \boldsymbol{\psi}(\tilde{\boldsymbol{d}}_N)\right)^T$ is an $(N \times (M_{\max} + 1))$-matrix, $\|\cdot\|_2$ denotes the Euclidean norm and the numbers L_{im} are defined by the roots

$$L_{im} := \left[\left(\sum_{\substack{|\boldsymbol{\alpha}|=1 \\ \boldsymbol{\alpha}=(\alpha_1,\alpha_2)^T}}^{2} \sum_{\substack{r<s \\ r,s\in V_m}} \left[D_{r,s}^{\boldsymbol{\alpha}} \psi_m(\hat{\boldsymbol{x}}_i^m)\right]^2 \right) \Delta \hat{\boldsymbol{x}}_i^m \right]^{1/2}. \tag{34}$$

For each derivative in (33) instead of using distinct penalty parameters we can use only one penalty parameter say $\lambda = \lambda_m$ $(:=\varphi^2)$ and write PRSS as follows [26]:

$$PRSS \approx \left\| \boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta} \right\|_2^2 + \lambda \left\| \boldsymbol{L}\boldsymbol{\theta} \right\|_2^2, \tag{35}$$

where $\boldsymbol{L}$ is a diagonal $(M_{\max} + 1) \times (M_{\max} + 1)$-matrix. Then our PRSS problem looks like a classical *Tikhonov regularization problem* with $\varphi > 0$, i.e., $\lambda = \varphi^2$ for some $\varphi \in \mathbb{R}$ [17].

We can tackle our Tikhonov regularization problem with *conic quadratic programming (CQP)* which is an optimization technique well-structured convex optimization [27]. We can formulate PRSS as a CQP problem based on an appropriate choice of a bound $\tilde{M}$ as follows [20]:

$$\begin{aligned} \min_{t,\boldsymbol{\theta}} \quad & t \\ \text{subject to} \quad & \left\| \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta} - \boldsymbol{y} \right\|_2 \leq t, \\ & \|\boldsymbol{L}\boldsymbol{\theta}\|_2 \leq \sqrt{\tilde{M}}. \end{aligned} \tag{36}$$

The values M in our optimization problem are determined by a model-free (train and error) method. When we access the M values in our CMARS code, CMARS provides us several solutions, each of them based on the basis functions.

Our CQP problem obeys the basic notation [16]

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} \quad & \|\boldsymbol{D}_i\boldsymbol{x} - \boldsymbol{d}_i\|_2 \leq \boldsymbol{p}_i^T\boldsymbol{x} - q_i \quad (i = 1, 2, \ldots, k). \end{aligned} \tag{37}$$

In fact, we can see that our optimization problem becomes such a CQP program with

$$\boldsymbol{c} = (1, \boldsymbol{0}_{M_{\max}+1}^T)^T, \qquad \boldsymbol{x} = (t, \boldsymbol{\theta}^T)^T, \qquad \boldsymbol{D}_1 = (\boldsymbol{0}_N, \boldsymbol{\psi}(\tilde{\boldsymbol{d}})), \qquad \boldsymbol{d}_1 = \boldsymbol{y}, \qquad \boldsymbol{p}_1 = (1, 0, \ldots, 0)^T,$$

$$q_1 = 0, \qquad \boldsymbol{D}_2 = (\boldsymbol{0}_{M_{\max}+1}, \boldsymbol{L}), \qquad \boldsymbol{d}_2 = \boldsymbol{0}_{M_{\max}+1}, \qquad \boldsymbol{p}_2 = \boldsymbol{0}_{M_{\max}+2} \quad \text{and} \quad q_2 = -\sqrt{\tilde{M}}.$$

For writing the optimality condition for this problem, we firstly reformulate the problem (36) as follows [19]:

$$\begin{aligned} \min_{t,\boldsymbol{\theta}} \quad & t \\ \text{such that} \quad & \boldsymbol{\chi} := \begin{bmatrix} \boldsymbol{0}_N & \boldsymbol{\psi}(\tilde{\boldsymbol{d}}) \\ 1 & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \begin{bmatrix} t \\ \boldsymbol{\theta} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{y} \\ 0 \end{bmatrix}, \\ & \boldsymbol{\eta} := \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1} & \boldsymbol{L} \\ 0 & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \begin{bmatrix} t \\ \boldsymbol{\theta} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1} \\ \sqrt{\tilde{M}} \end{bmatrix}, \\ & \boldsymbol{\chi} \in L^{N+1}, \qquad \boldsymbol{\eta} \in L^{M_{\max}+2}, \end{aligned} \tag{38}$$

where $L^{N+1}, L^{M_{\max}+2}$ are the $(N+1)$- and $(M_{\max}+2)$-dimensional second-order or Lorentz cones, defined by:

$$L^{N+1} := \left\{ \boldsymbol{x} = (x_1, x_2, \ldots, x_{N+1})^T \in \mathbb{R}^{N+1} \mid x_{N+1} \geq \sqrt{x_1^2 + x_2^2 + \cdots + x_N^2} \right\},$$

where $N \geq 1$.

The dual problem to the latter primal one is given by [20]:

$$\begin{aligned} \max_{y,\boldsymbol{\omega}_1,\boldsymbol{\omega}_2} \quad & (\boldsymbol{y}^T, 0)\boldsymbol{\omega}_1 + \left(\boldsymbol{0}_{M_{\max}+1}^T, -\sqrt{\tilde{M}}\right)\boldsymbol{\omega}_2 \\ \text{such that} \quad & \begin{bmatrix} \boldsymbol{0}_N^T & 1 \\ \boldsymbol{\psi}(\tilde{\boldsymbol{d}}) & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \boldsymbol{\omega}_1 + \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1}^T & 0 \\ \boldsymbol{L}^T & \boldsymbol{0}_{M_{\max}+1} \end{bmatrix} \boldsymbol{\omega}_2 = \begin{bmatrix} 1 \\ \boldsymbol{0}_{M_{\max}+1} \end{bmatrix}, \\ & \boldsymbol{\omega}_1 \in L^{N+1}, \qquad \boldsymbol{\omega}_2 \in L^{M_{\max}+2}. \end{aligned} \tag{39}$$

Moreover, $(t, \boldsymbol{\theta}, \boldsymbol{\chi}, \boldsymbol{\eta}, \boldsymbol{\omega}_1, \boldsymbol{\omega}_2)$ is a *primal dual optimal solution* if and only if [19]

$$\begin{aligned} & \boldsymbol{\chi} := \begin{bmatrix} \boldsymbol{0}_N & \boldsymbol{\psi}(\tilde{\boldsymbol{d}}) \\ 1 & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \begin{bmatrix} t \\ \boldsymbol{\theta} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{y} \\ 0 \end{bmatrix}, \\ & \boldsymbol{\eta} := \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1} & \boldsymbol{L} \\ 0 & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \begin{bmatrix} t \\ \boldsymbol{\theta} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1} \\ \sqrt{\tilde{M}} \end{bmatrix}, \\ & \begin{bmatrix} \boldsymbol{0}_N^T & 1 \\ \boldsymbol{\psi}(\tilde{\boldsymbol{d}}) & \boldsymbol{0}_{M_{\max}+1}^T \end{bmatrix} \boldsymbol{\omega}_1 + \begin{bmatrix} \boldsymbol{0}_{M_{\max}+1}^T & 0 \\ \boldsymbol{L}^T & \boldsymbol{0}_{M_{\max}+1} \end{bmatrix} \boldsymbol{\omega}_2 = \begin{bmatrix} 1 \\ \boldsymbol{0}_{M_{\max}+1} \end{bmatrix}, \\ & \boldsymbol{\omega}_1^T\boldsymbol{\chi} = 0, \qquad \boldsymbol{\omega}_2^T\boldsymbol{\eta} = 0, \\ & \boldsymbol{\omega}_1 \in L^{N+1}, \qquad \boldsymbol{\omega}_2 \in L^{M_{\max}+2}, \\ & \boldsymbol{\chi} \in L^{N+1}, \qquad \boldsymbol{\eta} \in L^{M_{\max}+2}. \end{aligned} \tag{40}$$

To solve "well-structured" convex programs like conic quadratic programming problems, interior point methods can be applied [27]. The interior-point optimizer is an implementation of the homogeneous and self-dual algorithm and it computes the interior point solution which is an arbitrary optimal solution [20]. The CMARS models are built up by using a special code written in MATLAB. In order to solve the CQP problem, the MOSEK interior-point optimization tool is preferred.

Table 1
Possible scenarios.

Observations	Score $< x$	Score $\geq x$
Default	True (A)	Misclassified (B)
Non-default	Misclassified (C)	True (D)

Table 2
Type of erros.

		Actual classes	
		Default	Non-default
Classes	Default	1-Type II	Type II
Specified by model	Non-default	Type I	1-Type I

2.2. Validation techniques

2.2.1. Receiver operating characteristic curve

The Receiver Operating Characteristic (ROC) curve is a tool that represents the possible distributions of scores for, in the case of our study, defaulting and non-defaulting countries. When determining the defaulting potentials of the countries, the decision maker should determine a cut-off value and classify the country as a potential defaulter when the score is lower than that cut-off value, or classify the country as a potential non-defaulter when a score is higher than the cut-off value. In such decisions, there are misclassifying mistakes.

There are true and wrong possible scenarios summarized in Table 1. The *hit rate* is defined as

$$HR(x) := \frac{H(x)}{N_D}, \tag{41}$$

where $HR(x)$ is the hit rate, $H(x)$ is the number of defaulter countries predicted correctly with the cut-off value x and N_D is the total number of defaulter countries in the sample. The *false alarm rate* is defined as:

$$FAR(x) := \frac{FR(x)}{N_{ND}}, \tag{42}$$

where $FAR(x)$ is the false alarm rate, $FR(x)$ is the number of non-defaulter countries predicted incorrectly as defaulters with the cut-off value x and N_{ND} is the total amount off non-defaulter countries.

The ROC curve is a graph of $HR(x)$ versus $FAR(x)$ drawn for each cut-off value. When comparing the scoring models, the model with the steeper ROC curve is better. However, just looking of the ROC curve can be misleading since it can be difficult to visualize the difference between curves. For this purpose, a *summary statistic* is needed. The most common summary statistic is the *Area Under the Curve (AUC)*. The worst model would have an AUC equal to 0.5, the best model has an AUC equal to 1.

2.2.2. Type of errors

The type of errors in a classification can be basically shown in Table 2.

Types of errors because of their easy calculations and interpretation properties, are the most widely used criterions in literature. For default prediction according to definition of errors, a Type II error is more important for researchers, since if such a misclassification is made there will be an actual loss for the creditors when a sovereign defaults.

2.2.3. 10-fold cross validation

Cross validation is a tool for measuring the model performance. In cross validation, some of the data are removed before training is applied. Then, when training is done, the remaining data are applied to test the performance of estimation. 10-fold cross validation is the most widely used method of cross validation. In 10-fold cross validation the data set is divided into 10 subsamples. Each time, one of the 10 subsamples is assumed as the validation sample and the other 9 samples are put together to form a training sample. Firstly, the training sample is employed to estimate model parameters then performance is evaluated by using the 10th subsample. Finally, the average performance measures for all 10 trials is computed.

3. Application

The definition of a sovereign default is a difficult subject of study because the decision of reconstruction of a country's debts is more a political issue rather than just a business decision. Therefore, it changes from government to government. In our study, the default definition is taken from [12]. It says, "A country is defined to be in debt crisis if it is classified as being in default by Standard and Poor's or if it receives a large non-concessional IMF loan defined as access in excess of 100 percent of quota" [12].

Table 3
List of determinants.

	Variables	Ratios and/or quatities (standardized)
	X_1	Bank liquid reserves to bank assets ratio (BLR)
	X_2	Current account balance (% of GDP) (CAB)
	X_3	Exports of goods and services (% of GDP) (EGS)
	X_4	GDP growth (annual %) (GDPG)
	X_5	Liquid liabilities (M3) as % of GDP (LL)
	X_6	Total debt service (% of exports of goods services and income) (TDS)
	X_7	Short-term debt (% of exports of goods services and income) (STD)
	X_8	Trade (% of GDP) (T)
	X_9	Inflation consumer prices (annual %) (ICP)
	X_{10}	Use of IMF credit/GDP (%) (IMF/GDP)
	X_{11}	Long-term debt/GDP (%) (LTD/GDP)
	X_{12}	External debt/total reserves (%) (ED/TR)

Table 4
Descriptive statistics of determinants.

	Sovereign default		Sovereign non-default	
	Average	Std. dev.	Average	Std. dev.
X_1	0.2070	1.1245	−0.1310	0.8891
X_2	0.0216	0.9756	−0.0136	1.0156
X_3	−0.1836	0.9493	0.1157	1.0145
X_4	−0.1664	0.9970	0.1049	0.9883
X_5	−0.2363	0.8459	0.1490	1.0599
X_6	−0.0880	0.2897	0.0555	1.2532
X_7	0.2332	1.1192	−0.1470	0.8868
X_8	−0.2069	0.9469	0.1305	1.0112
X_9	0.1311	1.4035	−0.0826	0.6108
X_{10}	0.6304	1.1920	−0.3974	0.5722
X_{11}	0.3631	0.9512	−0.2289	0.9622
X_{12}	0.3352	1.0279	−0.2113	0.9221

Table 5
Classification and prediction results of the models.

	Training sample			Validation sample		
	D–D (%)	ND–ND (%)	Overall correct classification rate (%)	D–D (%)	ND–ND (%)	Overall correct classification rate (%)
Logit	67.11	90.53	81.49	65.74	89.30	80.20
CART	92.88	96.82	95.30	65.99	83.39	76.67
GAM & CQP	74.99	88.94	83.55	71.57	87.38	81.27
MARS	88.32	92.41	90.83	75.38	82.91	80.00
CMARS	89.62	93.07	91.74	75.38	83.39	80.29

3.1. Data

Our sample contains 33 emerging markets in the period from 1980 to 2005. The sample includes 1019 yearly observations with 394 default cases according to our definition of sovereign default. For the variables, papers in literature account for the role of determinants in exposing a sovereign default. Liquidity, debt, economic stability, growth, and financial structure can be counted as important determinants of financial stability disabilities. Accordingly, we select 12 variables to determine the sovereign defaults. The names of determinants are listed in Table 3. We use the standardized values of determinants in our calculations. Descriptives of determinants for sovereign defaults are summarized in Table 4. According to these results, the big differences are shown in determinants for defaulting and non-defaulting sovereigns. Especially, in debt ratios average values of determinants showing debt play an important role on sovereign defaults. Growth and liquidity ratios also present differences.

3.2. Application results

In our applications, we apply 10-fold cross validation. According to the size of our data, we randomly select 102 observations for validation and we use remaining as a training sample. In the calculations, we employ MATLAB-2007.

In order to evaluate the classification capabilities of the models, the scoring results of the training and the validation samples are summarized in Table 5. It is inferred from Table 5 that, in the training sample, CART has a better classification capability than the other models in terms of the average correct classification rate. In the validation sample, GAM & CQP, CMARS and Logit approaches give more accurate results. Although GAM & CQP does not perform better than the other

Table 6
Classification and prediction results of the models.

	Training sample		Validation sample	
	Type I error	Type II error	Type I error	Type II error
Logit	0.0947	0.3289	0.1070	0.3426
CART	0.0318	0.0712	0.1661	0.3401
GAM & CQP	0.1106	0.2501	0.1262	0.2843
MARS	0.0791	0.1168	0.1709	0.2462
CMARS	0.0693	0.1038	0.1661	0.2462

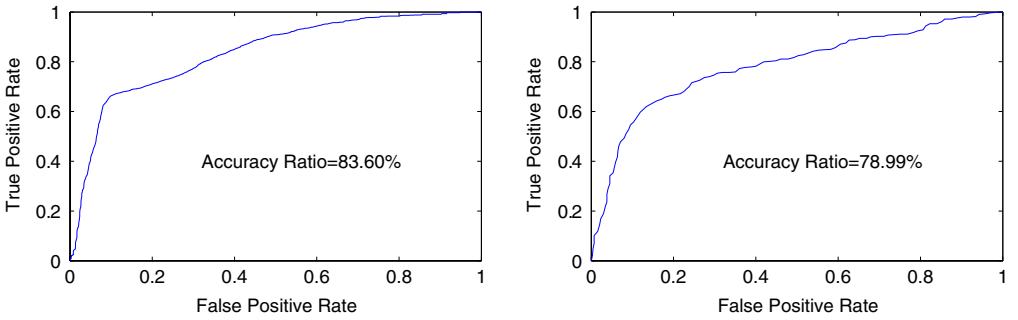

Fig. 1. ROC of logistic regression for training and validation samples, respectively.

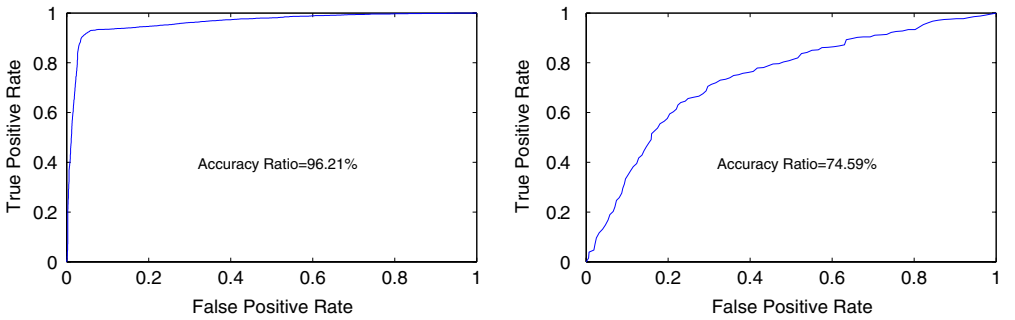

Fig. 2. ROC of CART for training and validation samples, respectively.

techniques in the training sample, it discovers the main structure of the data better than the other models and as a result it can be successfully applied in the validation sample after the model building procedure.

In order to justify the prediction performance of the designed models, we better evaluate the statistical error definitions. The misclassification probabilities have to be taken into account in order to obtain a model with the smallest expected misclassification costs in a statistical decision process. Hence, special attention also needs to be paid to Type I and Type II errors. Type I and Type II errors of the built models are summarized in Table 6.

According to Table 6, regarding default country assignation, MARS and CMARS have close Type II errors in comparison with each other. However, the other three models have higher Type II errors in validation which increase the serious risks associated with Type II errors. As a matter of fact, MARS and CMARS evidently detect default records better than the other models in validation. Moreover, in terms of Type I error, we can conclude that GAM & CQP and Logit have a better estimate compared with CART, CMARS and MARS. Error definitions show that CMARS and GAM & CQP return to be the best models in the determination of non-default country assignation.

In predicting country risk, since classification errors may create misleading results and wrong political decisions, it is important to evaluate the differentiating power of the models correctly. Let us recall that Receiver Operating Characteristic Curves (ROC) are one of the most popular validation techniques for evaluating classification success.

We look at discriminative power of the methods, respectively, from Figs. 1–5. First of all we observe big differences in the validation and training sample results for CART. Secondly, CART is only slightly better than CMARS, does not sustain its discrimination success in training sample and becomes the worst model in the validation sample results. Since CART is a discrete classification tool, it has difficulty in modelling an additive structure and the estimation variance increases with continuous explanatory variables. It can be concluded that CART is not suitable for future predictions compared to other techniques.

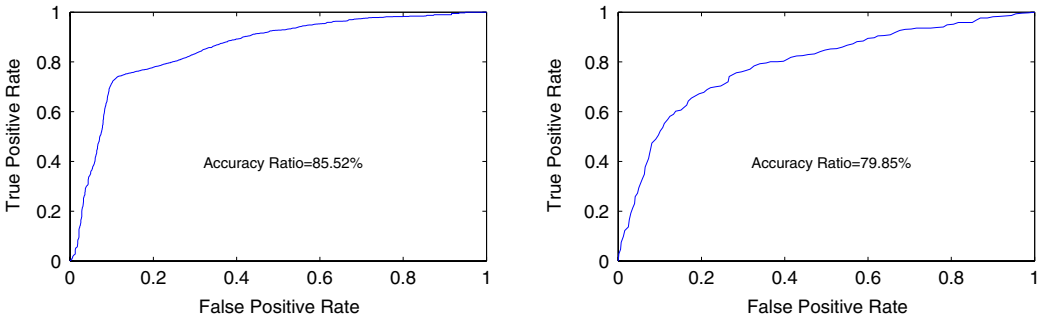

Fig. 3. ROC of GAM & CQP for training and validation samples, respectively.

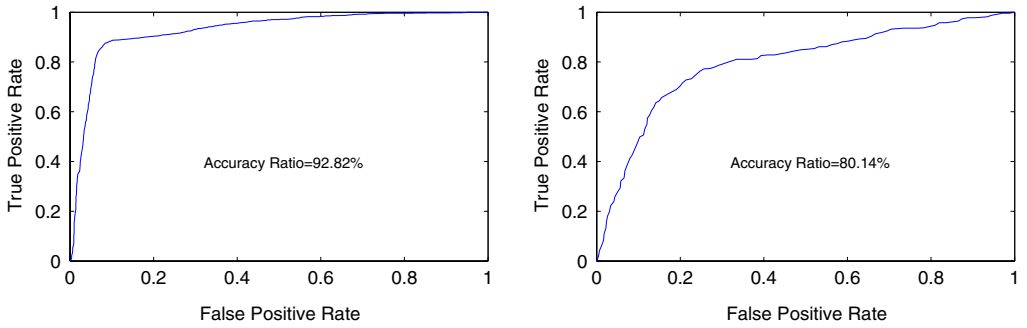

Fig. 4. ROC of MARS for training and validation samples, respectively.

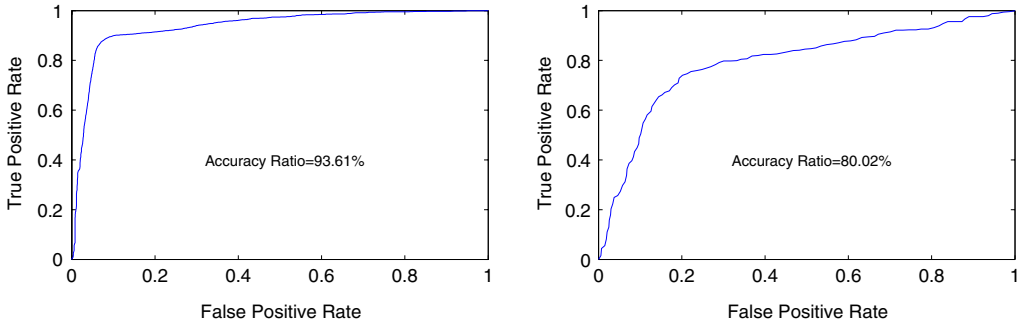

Fig. 5. ROC of CMARS for training and validation samples, respectively.

The GAM & CQP outperforms Logit and CART, but not as well as MARS and CMARS. The ROC Figs. 1–5 show that MARS and especially CMARS provide better accuracy results in validation samples. Actually, MARS uses piecewise linear functions which produce continuous models and provides a more effective way to model nonlinearities. MARS has the capability of determining the response variable making use of both additive and interaction effects of the predictors unlike CART and Logit.

On the other hand, CMARS provides its solutions by applying CQP. In this respect, it has the advantage of speed and complexity. CMARS permits the use of powerful interior point methods and is proved to be a useful tool for simplifying high-dimensional problems when there are many explanatory variables. CMARS enhances the classification power of the classical MARS and is robust enough for practical application, providing an efficient alternative for decision makers.

4. Conclusion

The increase in debt crises in emerging markets after 1980s aroused the interests of scientists in the area of sovereign defaults. In this study, we also consider the debt crisis, and we aim to capture the signals of debt crisis before crisis exists. In this work, we give an overview about the theoretical aspects of Logit, CART, and our GAM & CQP and MARS with their applications in estimating sovereign debt defaults.

This study also provides a new contribution to country default prediction by using modified MARS algorithm. The MARS algorithm is improved by constructing a penalized residual sum of squares (PRSS) as a Tikhonov regularization problem.

This problem is solved by using continuous optimization, especially, conic quadratic programming (CQP). We employ the program packages MOSEK, which is an optimization tool for solving large-scale mathematical optimization problems, Salford MARS and special codes written in MATLAB-2007, and provide an alternative modeling technique for MARS, named CMARS.

The results of the analysis show that CMARS outperforms traditional MARS in terms of the overall correct classification rate and, hence, it provides an efficient alternative in predicting default probabilities. CMARS provides its solutions by using CQP. In this respect, it has the advantage of speed and complexity permitting the use of powerful interior point methods.

The results also indicate that there is not a significant difference between CMARS and MARS solutions. However, CMARS is more robust than MARS and performance measures of CMARS show a higher stability. CMARS includes an improvement on the second part of the MARS algorithm, making the predictive accuracy of MARS better. CMARS seriously reduces the chance of making Type I error, compared with other methods. CMARS brings in a new point of view to the theory, methods and applications of mathematical data mining for displaying its success in financial theory by benefitting from modern continuous optimization.

The results of this research show that CMARS can be applied as much as MARS in the estimation of debt defaults and provides an efficient alternative for future research challenges in predicting sovereign defaults. GAM & CQP and the traditional Logit are also recommended as alternative tools having satisfactory estimation power.

In future studies, we shall include further new insights which we obtained in complexity reduction for CMARS [28] and our present development of a Robust CMARS (RCMARS), together with our colleagues. In fact, RCMARS will include uncertainty about future scenarios; especially, it will take into account stochasticity in the input variables X. Finally, we are working on new varieties of Generalized Partial Linear Models [29] which shall contain GAMs and CMARS as special cases. All these investigations will be conducted and applied in close collaboration with the practitioners in order to serve building up our countries and their financial markets.

References

[1] C.R. Frank, W.R. Cline, Measurement of debt servicing capacity, an application of discriminant analysis, Journal of International Economics 1 (1971) 327–344.
[2] C. Feder, R.E. Just, A study of debt servicing capacity applying logit analysis, Journal of Development Economics 4 (1977) 25–38.
[3] C. Fisk, F. Rimlinger, Nonparametric estimates of LDC repayment prospects, The Journal of Finance 34 (2) (1979) 429–436.
[4] R.J. Taffler, B. Abassi, Country risk: a model for predicting debt servicing problems in developing countries, Journal of the Royal Statistical Society 147 (1984) 541–568.
[5] E.M. Balkan, Political instability, country risk and probability of default, Applied Economics 24 (1992) 999–1008.
[6] J.C. de Haan, L.J. Siermann, E.V. Lubek, Political instability and country risk: new evidence, Applied Economics Letters 4 (1997) 703–707.
[7] P. Rivoli, T.L. Brewer, Political instability and country risk, Global Finance Journal 8 (1997) 309–321.
[8] J.C.B. Cooper, Artificial neural networks versus multivariate statistics: an application from economics, Journal of Applied Statistics 26 (8) (1999) 909–921.
[9] S.T. Easton, D.W. Rockerbie, What is in a default? Lending to LDCs in the face of default risk, Journal of Development Economics 58 (1999) 319–332.
[10] T.V. Gestel, B. Baesens, P.V. Dijcke, J. Garcia, J.A. Suykens, J. Vanthienen, A process model to develop an internal rating system: Sovereign credit ratings, Decision Support Systems 42 (2006) 1131–1151.
[11] D. Maltritz, A. Buhn, S. Eichler, Modeling country default risk as a latent variable: a multiple indicators multiple causes (MIMIC) approach, Spring Meeting of Young Economists, Istanbul, 2009.
[12] M. Fioramanti, Predicting sovereign debt crisis using artificial neural networks: a comparative approach, Preliminary version, 2006.
[13] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen, Classification and Regression Trees, 1st ed., Chapman and Hall, 1984.
[14] T. Hastie, R. Tibshirani, Generalized additive models: some applications, Journal of the American Statistical Association 82 (1987) 371–386.
[15] Copyright SAS, Generalized additive models, http://support.sas.com/rnd/app/da/new/dagam.html, 2009.
[16] P. Taylan, G.-W. Weber, A. Beck, New approaches to regression by generalized additive models and continuous optimization for modern applications in finance, Optimization 56 (2007) 1–24.
[17] P. Taylan, G.-W. Weber, F. Yerlikaya, I. Batmaz, G. Köksal, MARS algoritmasında Tikhonov düzenlemesi ve çok amaçlıoptimizasyon kullanımı, Preprint 105, IAM, METU, 2008. http://www3.iam.metu.edu.tr/iam/images/4/4f/Preprint105.pdf.
[18] J.H. Friedman, Multivariate adaptive regression splines, The Annals of Statistics 19 (1991) 1–141.
[19] P. Taylan, G-W. Weber, F. Yerlikaya, Continuous optimization applied in MARS for modern applications in finance, science and technology, in: ISI Proceedings of 20th Mini-EURO Conference Continuous Optimization and Knowledge-Based Technologies, 2008, pp. 317–322.
[20] F. Yerlikaya, A New Contribution to Nonlinear Robust Regression and Classification with Mars and its Applications to Data Mining for Quality Control in Manufacturing, Master Thesis, METU, 2008.
[21] Copyright Stat. Soft, Inc., Multivariate adaptive regression splines, http://www.statsoft.com/textbook/stmars.html, 2009.
[22] L. Chen, J. Song, F. Ji, MARS-based research of personal credit scoring: verification of Chinese data, Management Science and Engineering, International Conference on, 2006, pp. 1498–1501.
[23] W. Di, Long term fixed mortgage rate prediction using multivariate adaptive regression splines, Nanyang Technological University, 2006.
[24] D. Steinberg, P.L. Colla, K. Martin, MARS user guide, CA: Salford Systems, 1999.
[25] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Data Mining, Inference and Prediction, Springer, 2001.
[26] P. Craven, G. Wahba, Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation, Numerische Mathematik 31 (1979) 377–403.
[27] A. Nemirovski, Lectures on modern convex optimization, Israel Institute Technology, 2002. http://iew3.technion.ac.il/Labs/Opt/opt/LN/Final.pdf.
[28] I. Batmaz, F. Yerlika-Ozkurt, E. Kartal-Koc, G. Koksal, G.-W. Weber, Evaluating the CMARS performance for modeling nonlinearities, in: The Proceedings of PCO, 2010, Gold Coast, Australia, February 2–4, 2010.
[29] P. Taylan, G.-W. Weber, L. Liu, On foundations of parameter estimation for generalized partial linear models with B-splines and continuous optimization, in: The Proceedings of PCO, 2010, Gold Coast, Australia, February 2–4, 2010.